IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Application Transmittal

Assistant Commissioner for Patents
Washington, D.C. 20231

IBM DOCKET NO.YO998-137

IBM Corporation
Intellectual Property Law Department
P.O. Box 218
Yorktown Heights, New York 10598

Sir:

Date:April 30 , 1998

Transmitted herewith for filing is the **Patent Application** of:
Inventor(s):A. Dan et al
For:System and Method for Programmatic Generation of Continuous Media Presentations
Enclosed are:

[X] Sixteen (16) sheets of drawings. (INFORMAL)

[X] An assignment of the invention to International Busines Machines Corporation, Armonk, New York 10504.

[ ] A certified copy of a_____ application.

[X] Declaration and Power of Attorney

[ ] Information Disclosure Statement with copies of cited references. [ ] Associate Power of Attorney

The filing fee has been calculated as shown below:

	(Col. 1)	(Col. 2)	Other Than Small Entity	
For:	No. Filed	No. Extra	Rate	Fee
Basic Fee				$ 790.00
Total Claims	22 -20 =	2	x $22.00=	$ 44.00
Indep. Claims	4 - 3 =	1	x $82.00=	$ 82.00
[X] Multiple Dependent Claim Presented			$270.00	$ 270.00
			TOTAL	$ 1186.00

[XX] Please charge Deposit Account No 09-0468 in the amount of $1186.00. A duplicate copy of this sheet is enclosed.

[XX] The Commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account No. 09-0468. A duplicate copy of this sheet is enclosed.

[XX] Any additional filing fees required under 37 C.F.R §1 1

[XX] Any patent application processing fees under 37 C F R §1.17

Respectfully submitted,

BY: _____

Attorney David M. Shofi
Registration No.: 39,835 Tel.: (914)945-3247

"EXPRESS MAIL" LABEL NO. EM526133507
DATE OF DEPOSIT: April 30, 1998

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: A. Dan et al
Serial No.: Group No.
Filed: Herewith Examiner:
For: SYSTEM AND METHOD FOR PROGRAMMATIC GENERATION OF CONTINUOUS MEDIA PRESENTATIONS

Commissioner of Patents and Trademarks
Washington, D.C. 20231

EXPRESS MAIL CERTIFICATE

Express Mail Label Number EM526133507
Date of Deposit April 30, 1998

I hereby certify that the attached paper or fee

Patent Application Transmittal Letter (original and one copy)
Patent Application
Declaration and Power of Attorney
Informal Drawings (16 Sheets)
Recordation Form Cover Sheet
Assignment to IBM
Return Postcard

is being deposited with the United States Postal Service þExpress Mail Post Office to Addresseeþ service under 37 CFR 1.10 on the date indicated above and is addressed to the Commissioner of Patents and Trademarks, Washington, D.C. 20231.

Barbara Rasa
(Name)

Barbara Rasa
Signature

Note: Each paper must have its own certificate and the þExpress Mailþ label number as a part thereof or attached thereto. When, as here, the certification is presented on a separate sheet, that sheet must (1) be signed and (2) fully identify and be securely attached to the paper or fee it accompanies. Identification should include the serial number and filing date of the application as well as the type of paper being filed, e.g. complete application, specification and drawings, responses to rejection or refusal, notice of appeal, etc. If the serial number of the application is not known, the identification should include at least the name of the inventor(s) and the title of the invention.

Note: The label number need not be placed on each page. It should, however, be placed on the first page of each separate document, such as, a new application, amendment, assignment, and transmittal letter for a fee, along with the certificate of mailing by þExpress Mailþ. Although the label number may be on checks, such a practice is not required. In order not to deface formal drawings it is suggested that the label number be placed on the back of each formal drawing or the drawings be accompanied by a set of informal drawings on which the label number is placed.

Docket No. YO998-137

"EXPRESS MAIL" LABEL NO. EM526133507
DATE OF DEPOSIT: April 30, 1998

SYSTEM AND METHOD FOR PROGRAMMATIC GENERATION OF CONTINUOUS MEDIA PRESENTATIONS

FIELD OF THE INVENTION

This invention relates to the field of composition and presentation of multimedia. More specifically, the invention relates to the programmatic generation of multimedia presentation sequences.

BACKGROUND OF THE INVENTION

A block diagram of a conventional distributed multimedia presentation environment is shown in Fig. 1. A multimedia system in such an environment can produce a composed multimedia document (not shown) which is displayed on a presentation device 11, such as a television or computer monitor. First, various presentation documents 16, 17 are pre-composed by authors. This task of generating the presentation documents is called authoring. Many different types of documents can be authored, e.g., composition of video presentation 16 or composition of Web pages 17. Once these documents are prepared, they are stored in various data storage devices 14, 15 such as hard disks, digital video disks or storage devices of a satellite broadcasting company. Upon a user's request for presentation of a document, the presentation documents are delivered through the network 13 to the presentation device 11 for presentation. This is referred to as the "pull" mode. The presentation can also be sent by the storage devices 14, 15 pro-actively (e.g., by the broadcasting companies) to users without any explicit request via the network 13 or otherwise. This is referred to as the "push" mode.

Therefore, conventionally, multimedia presentation materials are generated before the presentation, i.e., at an authoring time. Once generated, presentation is accomplished exactly as the presentation materials are pre-composed. The user's capability to interact with the presentation is limited through interaction with the control panels 12 on the presentation device 11. Typical examples of the possible interactions provided by these control panels 12 include selection of different materials (channels or URLs), fast forward, fast reverse, pause, etc. However, in prior art systems, no dynamic changes to the presentation materials are supported.

A somewhat enhanced interaction capability is provided by the 3D object model. A pre-specified interaction semantic is built into the object. For example, PanoramIX, an image-based rendering software from IBM (www.software.ibm.com/net.media), uses environment maps (e.g., cylinders with tops/bottoms or polyhedral approximations of spheres) to render the background from a fixed viewpoint that does not translate. It supports smooth rotation and continuous zoom of the view. PanoramIX also allows pre-composition of a complex scene with embedded sprites, audio segments, video clips and 3D objects. It uses a control file that is pre-defined during scene authoring, using a special authoring tool.

Another example of a prior art system that uses pre-composed complex scenes is contained in the specifications of the MPEG-4 standard. MPEG-4 employs a BIFS (Binary Information for Scenes) composite file that describes a scene hierarchy, along the lines of VRML (Virtual Reality Modelling Language) scene graphs. Leaf nodes of the graph are AVOs (Audio-Visual Objects) that could be a video or audio or a 3D object or any other component media type. In fact, PanoramIX may be viewed as one instance of the MPEG-4 specification.

While they do provide some added interaction capability, these systems still fall short of providing the full ability to dynamically alter presentations after authoring.

SUMMARY OF THE INVENTION

The present invention provides a method by which a user can programmatically compose presentation materials which have been authored. First, users compose different presentations by applying dynamic synchronization, placement and selection of media stream segments under the control of a program. In contrast to pre-composed presentations where the selection of media segments and the synchronization and placement of those media segments are predetermined (e.g., MPEG-2 video), in the present invention, the selection, synchronization (for example, the determination of starting times for each media segment and delay insertion) and placements (in the display window) are controlled dynamically through the composition program. Second, the composition programs themselves are rule-based and event-driven, and therefore can be dynamically altered. Third, the program that controls composition may be downloaded or pre-resident at the presentation site. Additionally, various commands (understood by this program) may also be streamed to the program via separate logical channel(s). Finally, the composition can be accomplished both at the client site or at the server site based on user interactions at the client as well as network and server load.

Specifically, a method of dynamically generating a presentation sequence from a plurality of authored presentation documents is provided including the steps of receiving the plurality of authored presentation documents from a plurality of data sources, applying the

plurality of authored presentation documents to a set of presentation rules, and generating the presentation sequence in response to the applying step.

Preferably, the applying step includes the steps of testing for satisfied rule conditions and applying the plurality of authored presentation documents to a set of presentation rules in response to the satisfied rule conditions. In addition, it is preferable that the method include the step of receiving user input and wherein the generating step includes the step of generating the presentation sequence in response to the received user input. The method preferably includes the further step of modifying the set of presentation rules in response to the received user input.

The method can further include the step of sensing an external event, wherein the generating step includes the step of generating the presentation sequence in response to the sensed external event. Additionally, the method can further include the step of modifying the set of presentation rules in response to the sensed external even or in response to the received presentation documents.

The method can further include the steps of receiving meta data from the data sources and modifying the set of presentation rules in response to the received meta data.

The authored presentation documents are preferably authored presentation sequences and the generated presentation sequence is preferably a composite presentation sequence.

A method for programmatic generation of continuous multimedia presentations by a station capable of receiving at least one presentation and sensed events is also provided

including the steps of maintaining a library of rules, receiving at least one presentation; selecting at least one event to be sensed; receiving the event, testing each rule in the library for each received event; and optionally applying each rule to the presentation for each received event in response to the testing step to modify the presentation or to generate a new presentation.

A method for dynamically composing a presentation from a plurality of multimedia components is also provided including the steps of selecting one or more of the multimedia components to be identified as an initial portion of the presentation; programmatically selecting one or more other multimedia components to be identified as a subsequent portion of the presentation; disposing the subsequent portion with or following the initial portion; and synchronizing the selected components to form the presentation.

Preferably, this method further includes the step of presenting the presentation. The multimedia components can be continuous media components or non-continuous media components. They can also be audio or video components or text or image components.

It is preferable that the programmatically selecting step is responsive to input parameters, a past presentation history or a current state and to line content or meta-data.

A program storage device, readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for dynamically generating a presentation sequence according to the above-referenced method steps is also provided.

A programmatic presentation generator into which presentation documents are received is provided, including a rule base; a program state; and a computation engine, wherein the engine receives rule data from the rule base and state data from the program state and

generates a presentation sequence responsive to the presentation documents, the rule data and state data.

A set top box for receiving channels and sending presentation sequences to a digital television is also provided, including the programmatic presentation generator of the present invention.

Finally, a television set for receiving channels and generating presentation sequences is provided including the programmatic presentation generator of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Further objects, features and advantages of the present invention will become apparent from a consideration of the following detailed description of the invention when read in conjunction with the drawing figures, in which:

Fig. 1 is a block diagram of a conventional distributed multimedia presentation environment;

Fig. 1A is a block diagram of a presentation environment incorporating the programmatic presentation generator of the present invention;

Fig. 1B is a block diagram which shows one embodiment of the programmatic presentation generator of Fig. 1A incorporated as part of a set-top box;

Fig. 1C is a block diagram showing an Intelligent TV Guide example of using the programmatic presentation generator of the present invention;

Fig. 1D is a block diagram showing an intelligent slide show example of using the

programmatic presentation generator of the present invention;

Fig. 2 is a block diagram of the presentation environment of Fig. 1A showing the framework of the programmatic presentation generator according to the present invention;

Fig. 3 is a block diagram of a rule of the programmatic presentation generator framework of the present invention;

Fig. 4 is a flowchart of the process to generate a presentation sequence according to the present invention;

Fig. 5 is a block diagram of a conventional presentation sequence;

Fig. 6 is a block diagram of a conventional static object;

Fig. 7 is a block diagram of a composite presentation sequence of the present invention;

Fig. 8 and Fig. 8A are block diagrams showing the operations to define and change the temporal properties of a presentation sequence according to the present invention; and

Fig. 9, Fig. 9A and Fig. 9B are block diagrams for the operations to define and change the temporal properties of a composite presentation sequence using multiple presentation sequences according to the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Figure 1A is a block diagram of a distributed multimedia presentation environment utilizing the programmatic presentation generator of the present invention. In Fig. 1A, a presentation device 1A1, such as a TV or computer monitor, presents a presentation sequence

1A5 generated by the programmatic presentation generator 1A4. This programmatic presentation generator 1A4 can generate the presentation sequences using the data pushed from data source 1A3 or it can pull the data from the data source 1A2 when required. These data sources can include a hard disk, digital video disk, satellite or TV broadcasting company. The programmatic presentation generator 1A4 according to the present invention permits a presentation sequence, such as TV programs, MPEG sequences, etc., to be dynamically generated or, once generated, to be dynamically modified reflecting user input, presentation rules and external events.

This programmatic presentation generator 1A4 could be incorporated on different software components. For example, it can be incorporated as part of the set-top box environment or executed as part of the server software.

Fig. 1B shows the block diagram of a programmatic presentation generator 1B5 used in a set-top box 1B2. A television 1B1 receives the presentation sequences from a set-top box 1B2. Alternately, one skilled in the art can appreciate that the programmatic presentation generator can be located in the television 1B1 itself, assuming the functionality of the set-top box 1B2 is incorporated therein. In prior art systems, users could only select a channel of interest out of the multiple incoming channels 1B4. However, by placing the programmatic presentation generator 1B5 in a set-top box 1B2 (or in a television), users can dynamically modify or compose a presentation sequence using the multiple incoming channels 1B4. The components of a programmatic presentation generator 1B5 (i.e, rule base, engine, program state) are described hereinbelow.

Fig. 1C is a block diagram which shows an Intelligent TV Guide as an example

using the programmatic presentation generator of the present invention. Multiple channels 1C1 are connected to a TV 1C3. Through certain channels 1C2, additional meta (command) data can also be transmitted. Using the programmatic presentation generator of the present invention, a user can watch the programs in which he is most interested. This selection could be specified using rule-based programs as described hereinbelow. This program could use simple keyword matching, similar to that used in an internet search engine, over the meta data. For example, a program could be added to the rule-base (described hereinbelow) so that automatic selection of the contents is executed every 30 minutes. As an example, the rules can be specified in the following way: First, repeated selection of the same content should be avoided. Second, violent programs may not be selected. Also, if a new interesting program shows up on the display and if it is within the threshold of preference (i.e., evaluated via a user-defined function as sufficiently interesting), the next two best programs are shown in the top corner windows 1C4, 1C5 for 2 minutes. This functionality significantly assists a user with the task of choosing a program to view.

Fig. 1D is a block diagram which shows an Intelligent Slide Show as another example of an application of the programmatic presentation generator of the present invention. The composed, original slide show 1D1 consists of 4 slides (1D21 through 1D24). Two of the slides (1D21 and 1D23) include URLs. The slide show 1D1 was designed to be played back in the regular playback order shown in 1D2, i.e., in the order of 1D21, 1D22, 1D23 and 1D24. A different slide show can be generated using the programmatic presentation generator of the present invention as shown in 1D3. The new (extended playback) slide show 1D3 has been

generated by extending 1D1. In the slide show 1D3, the contents of the URLs included in the original slides (1D21 and 1D23) are retrieved and included as parts of the slides. The inclusion of the pages referenced by the URLs can be accomplished in various ways. For example, the slide 1D21 was extended such that the referenced URL page 1D312 is included as a part of the same slide as page 1D311 as shown in 1D31. In contrast, the referenced URL page on slide 1D23 is included as a separate slide 1D34.

Fig. 2 is a diagram which shows a more detailed view of the programmatic presentation generator according to the present invention along with a distributed multimedia presentation environment. Data sources 21, 22, 23, 24 and network 25 are the same as those in Fig. 1. The framework 26 of the programmatic presentation generator is composed of a rule-base 261, a presentation state 262 and a computation engine 263.

The rule-base 261 is composed of a set of rules to be described hereinbelow. The rule-base 261 can be modified (i.e., adding, deleting or modifying one or more rules) in several ways. First, modification can occur by local user input. Second, one or more existing rules can implicitly modify the rule-base. Finally, data contained in the presentation sequence(s) received from the data source(s) or meta data received from the data source(s) in a separate logical channel (whether initiated by a user pull command or by a data source push command) can alter the rule-base.

The presentation state 262 consists of a presentation history and a set of state variables. The presentation history includes the history of such information as presentation contents, user interactions and external events up to a certain point in time. A state variable is a

program-defined variable which serves to mark and represent the status of a presentation so that it can be referred within a program (rules).

The computation engine 263 includes computational logic which monitors the presentation state 262, interprets the rule base 261 and sensed external events, and modifies the presentation sequence, thereby generating a new presentation sequence 264 and a resulting presentation 27. The sensed external events can include input timing parameters and user inputs. Upon each sensed external event, the engine 263 serializes the rules in the rule-base 261, and applies each rule to the current presentation sequence. That is, given a rule-base RB = $\{R_1, R_2, \ldots, R_n\}$, where R_i is a rule, and the current presentation sequence P, the engine generates the new presentation sequence P_n as follows:

$$P==\{R_1\}==>P_1==\{R_2\}==>P_2==\{R_3\}==>P_3==\{R_n\}==>P_n$$

where P_i is the presentation sequence generated from the presentation sequence P_{i-1} by applying rule R_i.

Fig. 3 shows a block diagram of a rule within the framework of the programmatic presentation generator of the present invention. A rule is a pair of TEST 31 and ACTION 32. TEST 31 specifies the condition 311 to be met for the ACTION to be fired. It can be any predicate over the presentation states such as "if at time t_i, a specific image has been presented."

ACTION 32 is a sequence of operations on a presentation sequence to generate a new presentation. These operations can be realized by any code segments using any programming language such as C, PASCAL, JAVA, etc. The operations in an ACTION are preferably composed of four parts 321, 322, 323, 324. The first code segment 321 is to specify the selection

of different content data to be included in the presentation. The second is the code segment 322 which specifies different temporal orders of presentation. This code segment uses the operations which will be described in Figs. 8, 8A, 9, 9A and 9B. The third code segment 323 is to specify the different spatial layouts among different presentation content data. Lastly, there can be code segments 324 for changing presentation attributes, e.g., quality, color, etc. In addition, code segments to change the presentation states 325 (e.g., by utilizing state variables) can be interleaved with the above-mentioned operations. Here, annotations in the input presentation can be monitored and state variables can be adjusted accordingly.

In the application of a rule R_i = <TEST,ACTION> to a presentation sequence P, first, it is tested if the condition TEST is met and if so, the code segment ACTION is applied to the presentation sequence P. More mathematically, when the condition TEST is satisfied (is TRUE), an ACTION can be specified as P_{old} ==ACTION==>P_{new} where P_{old} is the current presentation sequence and P_{new} is the new presentation sequence.

Fig. 4 is a flowchart illustrating a preferred method for the computation engine 263 to generate a new presentation sequence upon detection of an event. In the diagram, the current presentation is represented by P, the rule-base by $\{R_1, R_2, \ldots, R_n\}$, each rule in the rule-base by R_i, the number of rules in the rule-base by n, and the new presentation generated at each computation step by P_{new}. In step 41, a new event is sensed and the number of rules, n, in the rule base is received. Next, the engine initializes the index variable i to 0 and initializes the new presentation sequence P_{new} to the current presentation sequence P in step 42. Then, in step 43, the index variable i is increased by 1. In step 44, the index variable i is compared to the number of

rules, i.e., n, in the rule-base. If the index variable is greater than the number of rules, the current P_{new} is output as a new current presentation by setting $P = P_{new}$. Otherwise, in step 45, the *i*th rule, R_i is selected and is tested. If the TEST of R_i is not TRUE, the method proceeds in step 43. Otherwise, if the test of R_i is TRUE, in step 46, the ACTION of R_i is applied to P_{new} to update P_{new}. Then, the method continues with step 43.

Fig. 5 shows a block diagram of a presentation sequence D. The presentation sequence D is composed from a sequence of static objects 51, 52, 53, 54. . ., etc. and then by assigning the relative time t_k to each static object 51, 52, 53, 54 . . ., etc. in the presentation sequence. A static object is a presentation object without any temporal property. Examples of the static objects include an image, text, a video frame, etc.

Fig. 6 shows a block diagram of a composite static object. A composite static object is composed by spatially relating multiple composite objects 61 and 62 on a screen. The description of the spatial relation can be accomplished by using techniques known in the prior art. One such technique can be found in IBM Research Report, Hyperstory, M. Kim and J. Song, 1995.

Fig. 7 shows a block diagram of a composite presentation sequence. A composite presentation sequence consists of two or more presentation sequences. The composite presentation sequence in Fig. 7 consists of three presentation sequences D1, D2 and D3.

The operation to create a presentation sequence can be represented by Create, i.e.,

$$\text{Create } (< (s_1, t_1), (s_2, t_2), \ldots, (s_n, t_n) >).$$

Given a sequence of static objects, $s_1, s_2, \ldots, s_n$, a presentation sequence $< (s_1, t_1), (s_2, t_2), \ldots, (s_n,$

t_n) > is created in which static object s_i is displayed during virtual time interval (t_i, t_{i+1}). For example, in Fig. 5, the static object 51 is shown during the virtual time interval (t_1, t_2), and static object 52 is shown during virtual time interval (t_2, t_3), etc.

A presentation sequence can also be generated from one or more existing presentation sequences using temporal operators. A set of temporal operations for the description of temporal property can also be found in "Composition and Search with Video Algebra" by Ron Weiss, Andrej Duda and David K. Gifford in IEEE Multimedia, 1995.

Fig. 8 and Fig. 8A illustrate different operations on a presentation sequence according to the present invention. The intra-sequence operations include:

Head (D,i) 81: Given a presentation sequence D = <(s_1, t_1), (s_2, t_2), . . ., (s_n, t_n) >, a subsequence ending at the *i*th object is selected as a new presentation sequence. That is, D_{new} = <(s_1, t_1), (s_2, t_2),. . ., (s_i, t_i) >.

Tail (D,i) 82: Given a presentation sequence D = < (s_1, t_1), (s_2, t_2),. . .(s_n, t_n) >, a subsequence starting from the *i*th object is selected as a new presentation sequence. That is, D_{new} = < (s_i, t_i), (s_{i+1}, t_{i+1}),. . .,(s_n,t_n) >.

Delay (D,t) 83: Given a presentation sequence D = < (s_1, t_1),(s_2, t_2),. . .(s_n, t_n) >, a new presentation sequence D_{new} is created in which the virtual display times are delayed by t virtual time units. That is, D_{new} = < (s_1, t_1 + t), (s_2, t_2 +t),. . ., (s_n, t_n + t) >.

Stretch (D,r) 84: Given a presentation sequence D = <(s_1, t_1), (s_2, t_2),. . ., (s_n, t_n) >, a new presentation sequence D_{new} is created in which the starting times t_i of static object s_i is scaled to t_i x r, i.e., D_{new} = <(s_1, t_1 x r), (s_2, t_2 x r), . . .,(s_n, t_n x r)>.

Repeat (D,i,j,r) 85: Given a presentation sequence D, a new presentation sequence is created in which the subsequence from s_i to s_j is repeated r times.

A composite presentation sequence also can be generated by temporally relating multiple presentation sequences (i.e., via inter-sequence operations) according to the present invention. Fig. 9, Fig. 9A, and Fig. 9B show the block diagrams for such inter-sequence temporal operations. They include:

Meet (D_1, D_2) 91: Given two presentation sequences D_1 and D_2, a new presentation sequence D_{new} is created in which the starting time of D_2 is the same as the ending time of D_1.

Co-Start (D_1, D_2) 92: Given two presentation sequence D_1 and D_2, a new presentation sequence D_{new} is created in which the starting time of D_1 and D_2 are the same.

Co-End (D_1, D_2) 93: Given two presentation sequences D_1 and D_2, a new presentation sequence D_{new} is created in which the ending time of D_1 and D_2 are the same.

Interleave (D_1, D_2, d) 96: Given two presentation sequences, D_1 and D_2, a new presentation sequence is created in which the two sequences are interleaved. Here the parameter d represents the delay in the new presentation sequence of the start time of D_2 relative to the start time of the presentation sequence D_1.

Given a presentation sequence, a presentation is constructed by associating a start time and a play rate. More mathematically, $P = <D, t_s, r>$ is a presentation where the presentation sequence D is started at real time S with play rate r. Here the start time t_s and the play rate r are used to map virtual times to real times. For example, virtual time t_i is mapped to real time

$t_s + t_i \times r$.

Thus, an important feature of the framework of a programmatic generation of a presentation according to the present invention is that the rule-base can be dynamically changed by inserting/deleting/modifying rules and the changes can be dynamically reflected in the generation of a presentation with the engine.

The framework of the programmatic generation of the presentation sequence according to the present invention can be made in either the multimedia server or in the client presentation system or both.

Now that the invention has been described by way of a preferred embodiment, various modifications and improvements will occur to those of skill in the art. Thus, it should be understood that the preferred embodiment is provided as an example and not as a limitation. The scope of the invention is defined by the appended claims.

WE CLAIM:

1. A method of dynamically generating a presentation sequence from a plurality of authored presentation documents comprising the steps of:

receiving the plurality of authored presentation documents from a plurality of data sources;

applying the plurality of authored presentation documents to a set of presentation rules; and

generating the presentation sequence in response to the applying step.

2. The method of claim 1 wherein the applying step comprises the steps of:

testing for satisfied rule conditions; and

applying the plurality of authored presentation documents to a set of presentation rules in response to the satisfied rule conditions.

3. The method of claim 2 further comprising the step of receiving user input and wherein the generating step comprises the step of generating the presentation sequence in response to the received user input.

4. The method of claim 3 further comprising the step of modifying the set of presentation rules in response to the received user input.

5. The method of claim 2 further comprising the step of sensing an external event and wherein the generating step comprises the step of generating the presentation sequence in response to the sensed external event.

6. The method of claim 5 further comprising the step of modifying the set of presentation rules in response to the sensed external event.

7. The method of claim 1 further comprising the step of modifying the set of presentation rules in response to the received presentation documents.

8. The method of claim 1 further comprising the steps of:

receiving meta data from the plurality of data sources; and

modifying the set of presentation rules in response to the received meta data.

9. The method of claim 1 wherein the authored presentation documents are authored presentation sequences and the generated presentation sequence is a composite presentation sequence.

10. A method for programmatic generation of continuous multimedia presentations by a station capable of receiving at least one presentation and a plurality of sensed events, the method comprising the steps of:

maintaining a library of rules;

receiving at least one presentation;

selecting at least one event to be sensed;

receiving the at least one event;

testing each rule in the library for each received event; and

optionally applying each rule to the at least one presentation for each received event in response to the testing step to modify the at least one presentation or to generate a new presentation.

11. A method for dynamically composing a presentation from a plurality of multimedia components comprising the steps of:

selecting one or more of the multimedia components to be identified as an initial portion of the presentation;

programmatically selecting one or more other multimedia components to be identified as a subsequent portion of the presentation;

disposing the subsequent portion with or following the initial portion; and

synchronizing the selected components to form the presentation.

12. The method of claim 11 further comprising the step of presenting the presentation.

13. The method of claim 11 wherein the multimedia components are continuous media

components.

14. The method of claim 13 wherein the multimedia components are audio or video components.

15. The method of claim 11 wherein the multimedia components are non-continuous media components.

16. The method of claim 15 wherein the multimedia components are text or image components.

17. The method of claim 11 wherein the programmatically selecting step is responsive to input parameters, a past presentation history or a current state.

18. The method of claim 17 wherein the programmatically selecting step is responsive to line content or meta-data.

19. A program storage device, readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for dynamically generating a presentation sequence according to the method steps of claims 1-18.

20. A programmatic presentation generator into which presentation documents are received, comprising:

a rule base;

a program state; and

a computation engine;

wherein the engine receives rule data from the rule base and state data from the program state and generates a presentation sequence responsive to the presentation documents, the rule data and state data.

21. A set top box for receiving channels and sending presentation sequences to a digital television, comprising the programmatic presentation generator of claim 20.

22. A television set for receiving channels and generating presentation sequences, comprising the programmatic presentation generator of claim 20.

ABSTRACT OF THE DISCLOSURE

A method of dynamically generating a presentation sequence from a plurality of authored presentation documents includes the steps of receiving the plurality of authored presentation documents from a plurality of data sources; applying the plurality of authored presentation documents to a set of presentation rules; and generating the presentation sequence in response to the applying step.

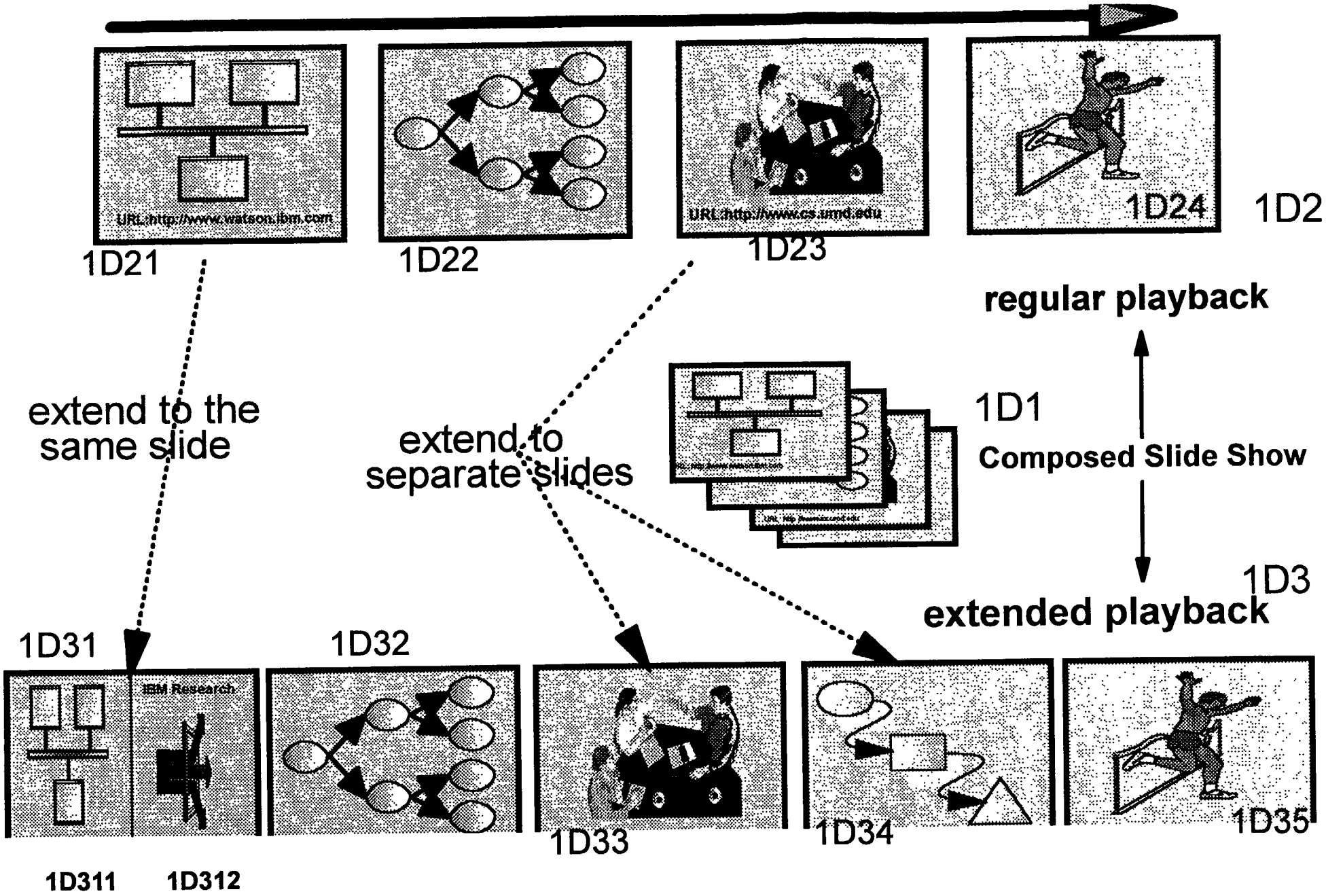

URL:http://www.watson.ibm.com
URL:http://www.cs.umd.edu
1D24
1D2
1D21
1D22
1D23
regular playback
extend to the same slide
extend to separate slides
1D1
Composed Slide Show
1D3
extended playback
1D31
1D32
IBM Research
1D33
1D34
1D35
1D311
1D312

FIG. 1D

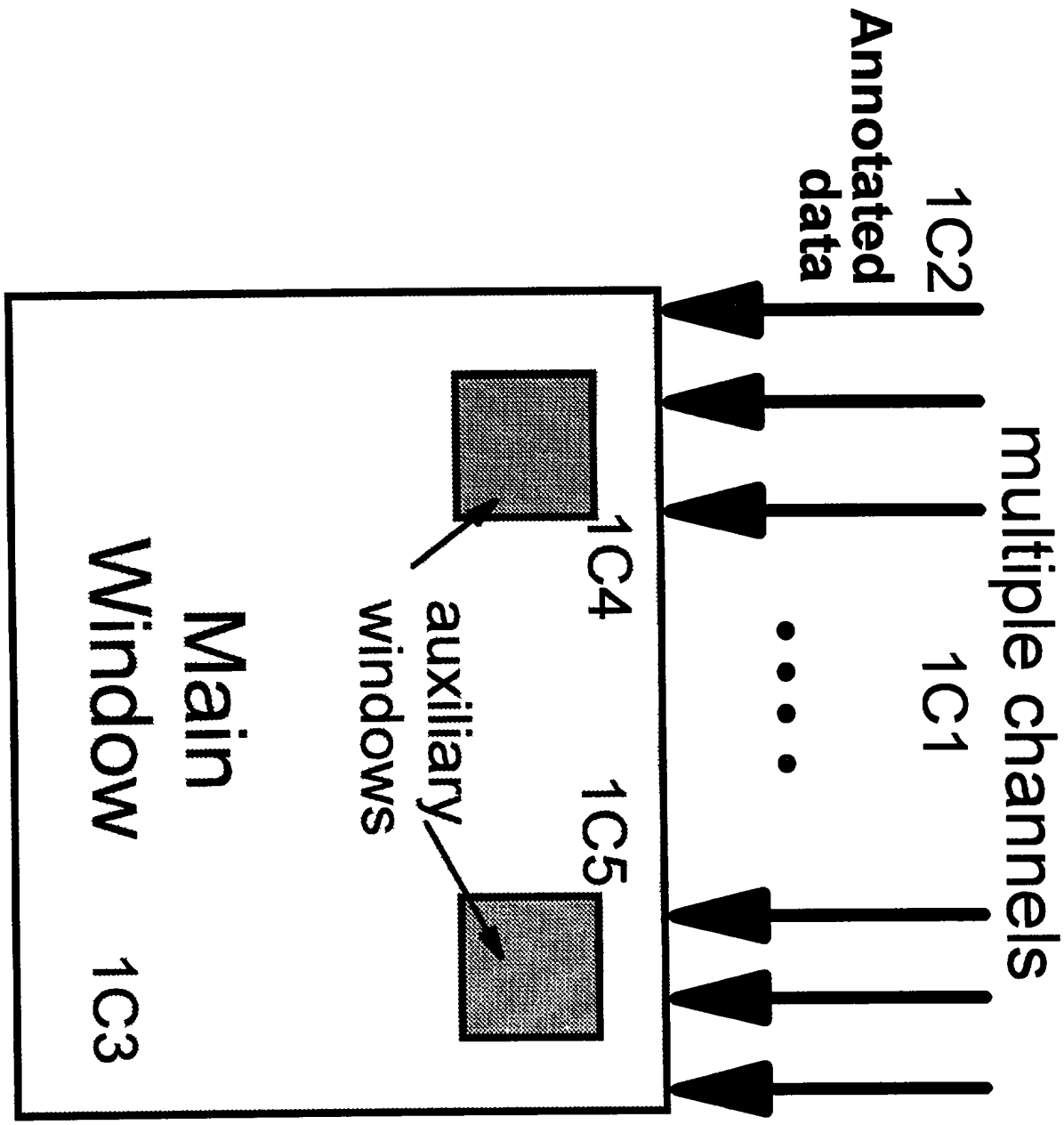

multiple channels
1C1
1C2
Annotated data
1C4
1C5
auxiliary windows
Main Window
1C3

FIG. 1C

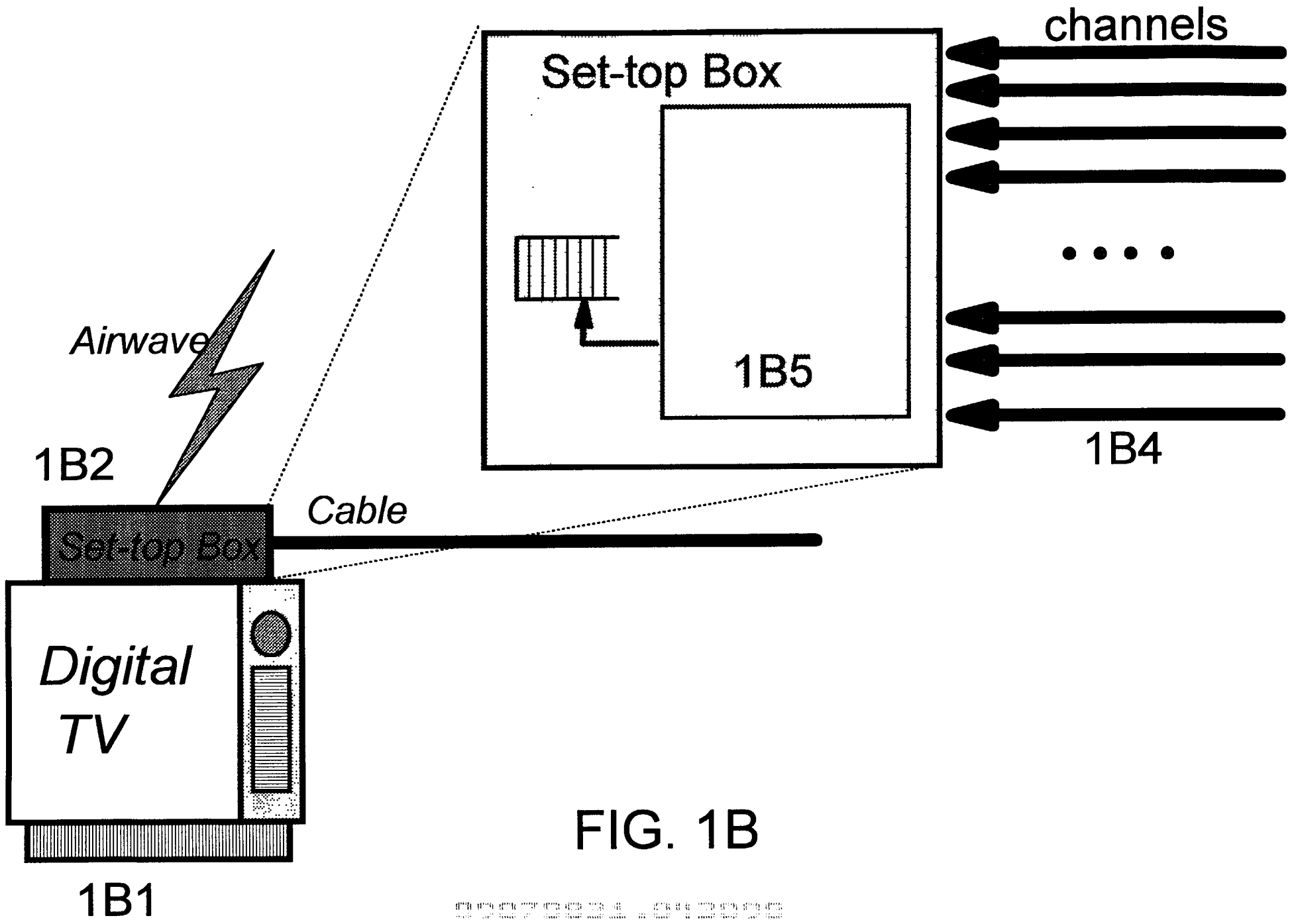

channels
Set-top Box
1B5
1B4
Airwave
1B2
Cable
Set-top Box
Digital
TV
1B1

FIG. 1B

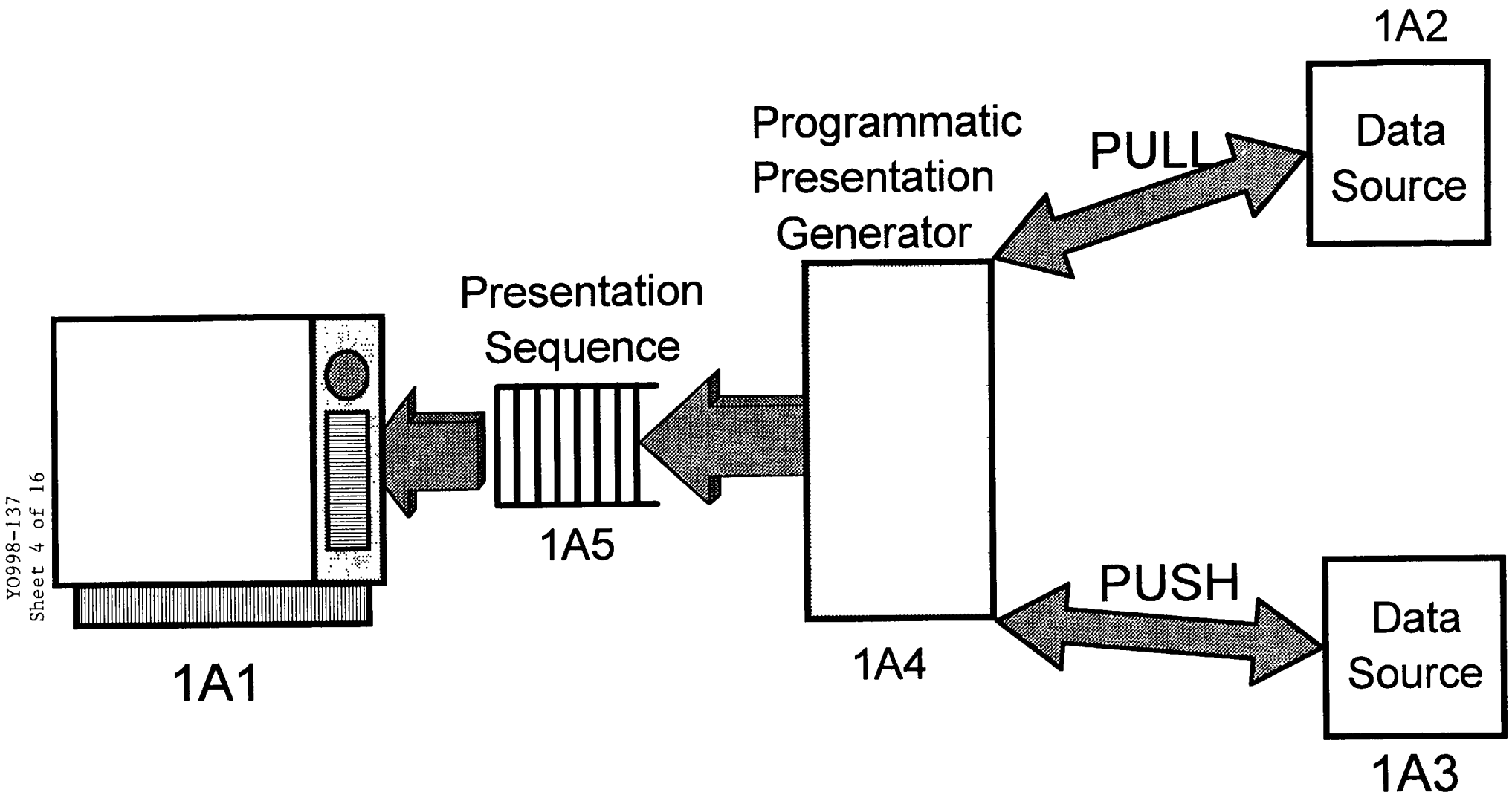

1A2
Data Source
Programmatic Presentation Generator
PULL
Presentation Sequence
1A5
1A1
1A4
PUSH
Data Source
1A3

FIG. 1A

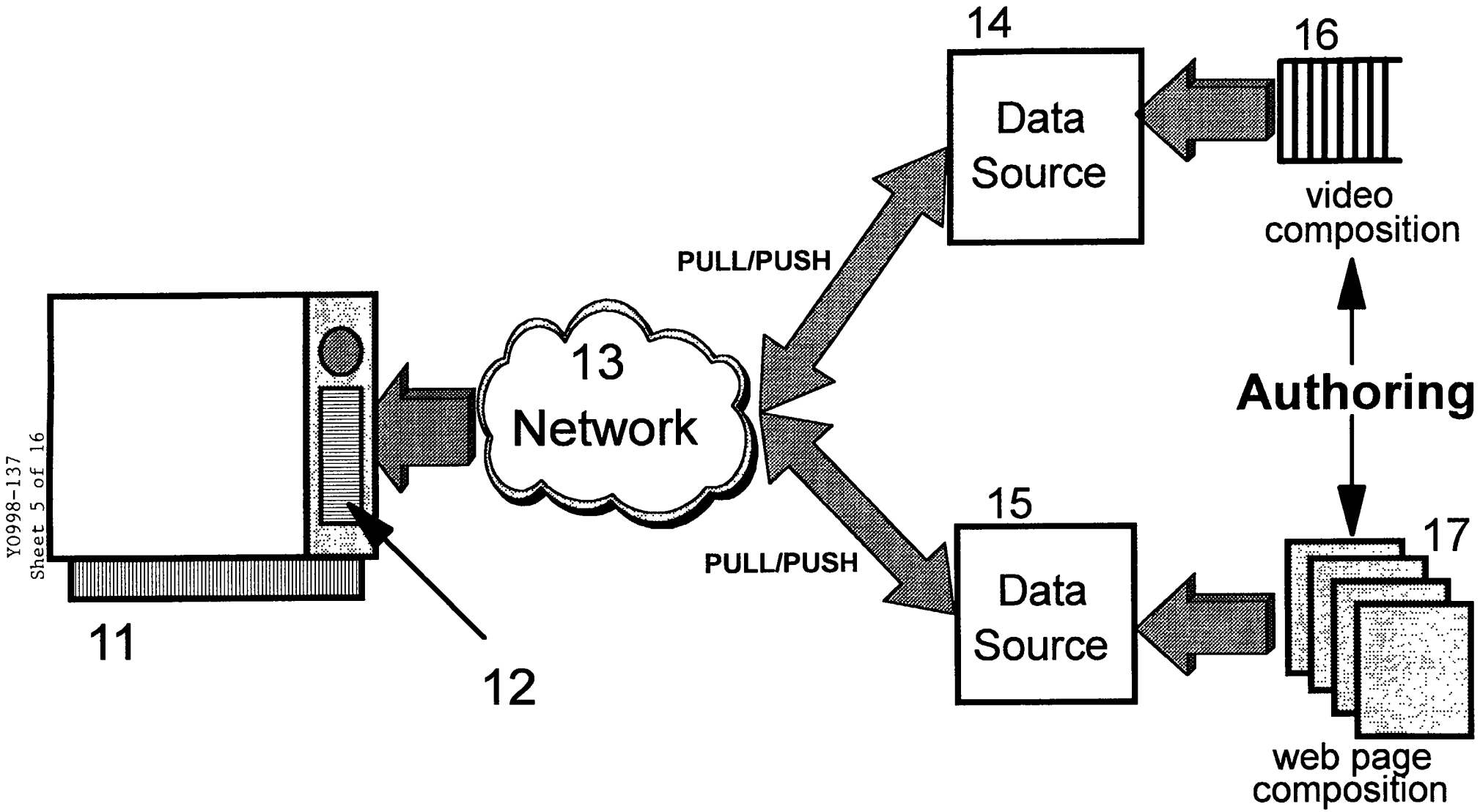


FIG. 1

(PRIOR ART)

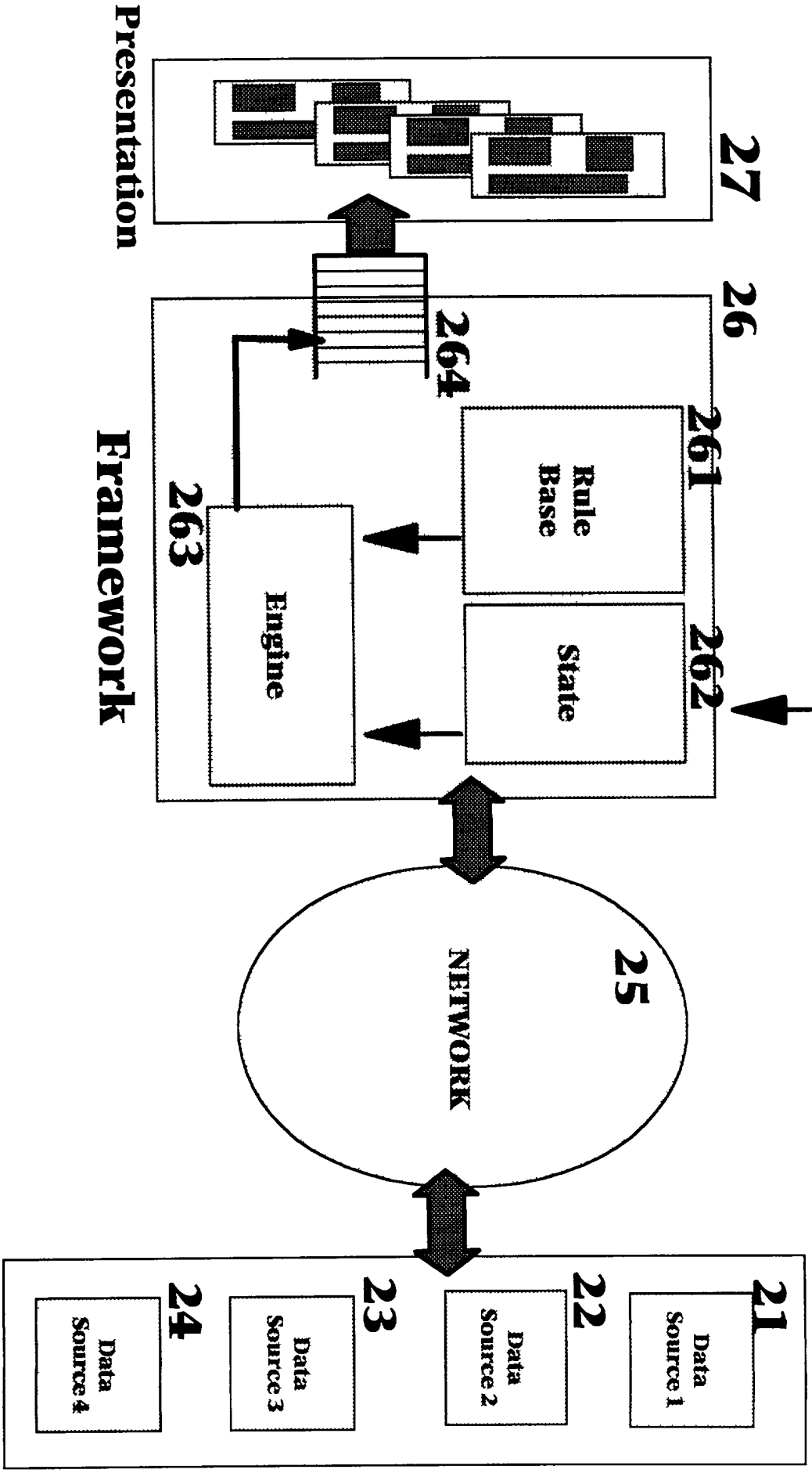

Presentation
27
26
264
261
Rule Base
263
Engine
262
State
Framework
External Events
25
NETWORK
21
Data Source 1
22
Data Source 2
23
Data Source 3
24
Data Source 4

FIG. 2

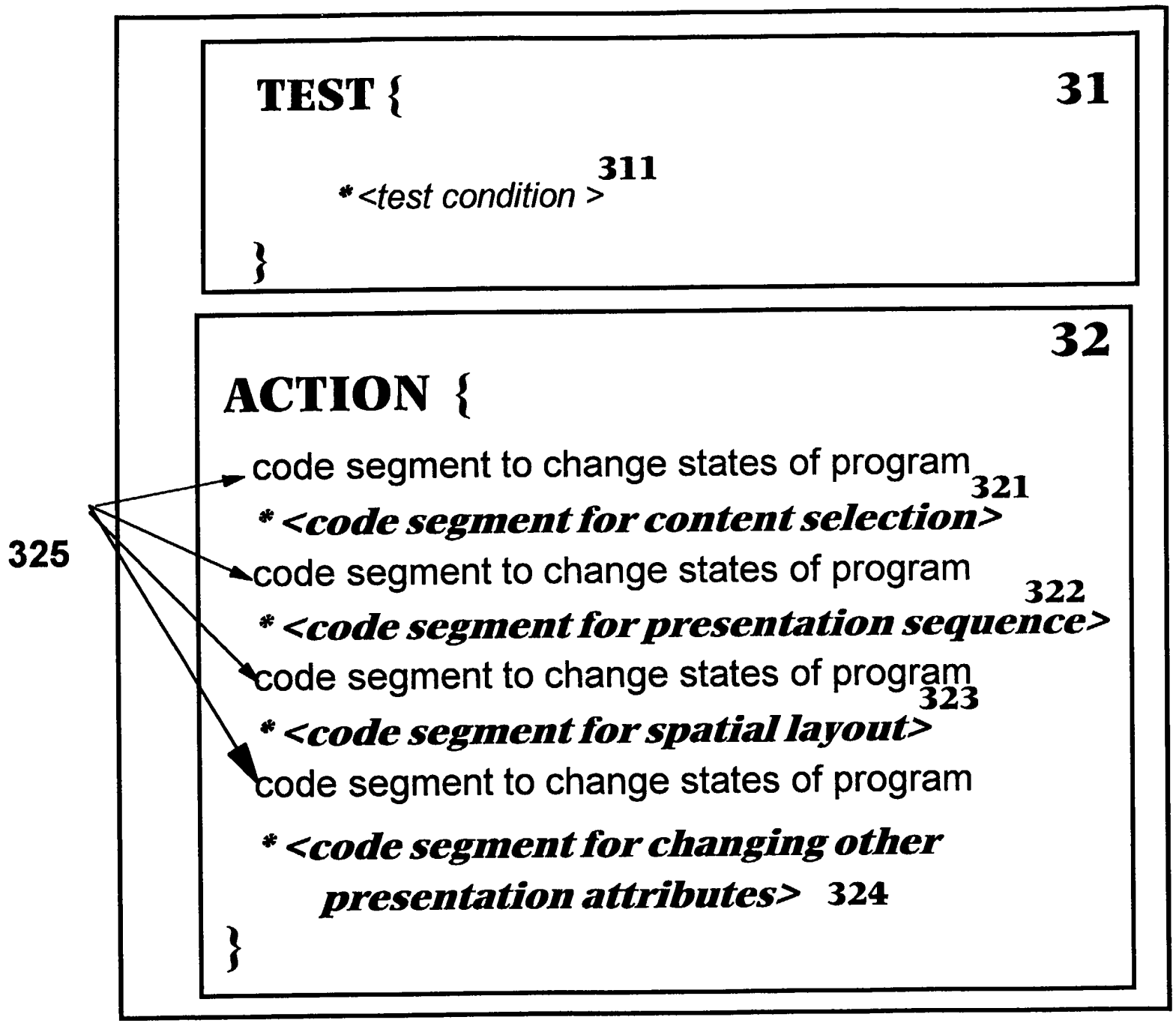

TEST {
31
* <test condition >
311
}
32
ACTION {
code segment to change states of program
321
* <code segment for content selection>
code segment to change states of program
322
* <code segment for presentation sequence>
code segment to change states of program
323
* <code segment for spatial layout>
code segment to change states of program
* <code segment for changing other presentation attributes>
324
}
325

FIG. 3

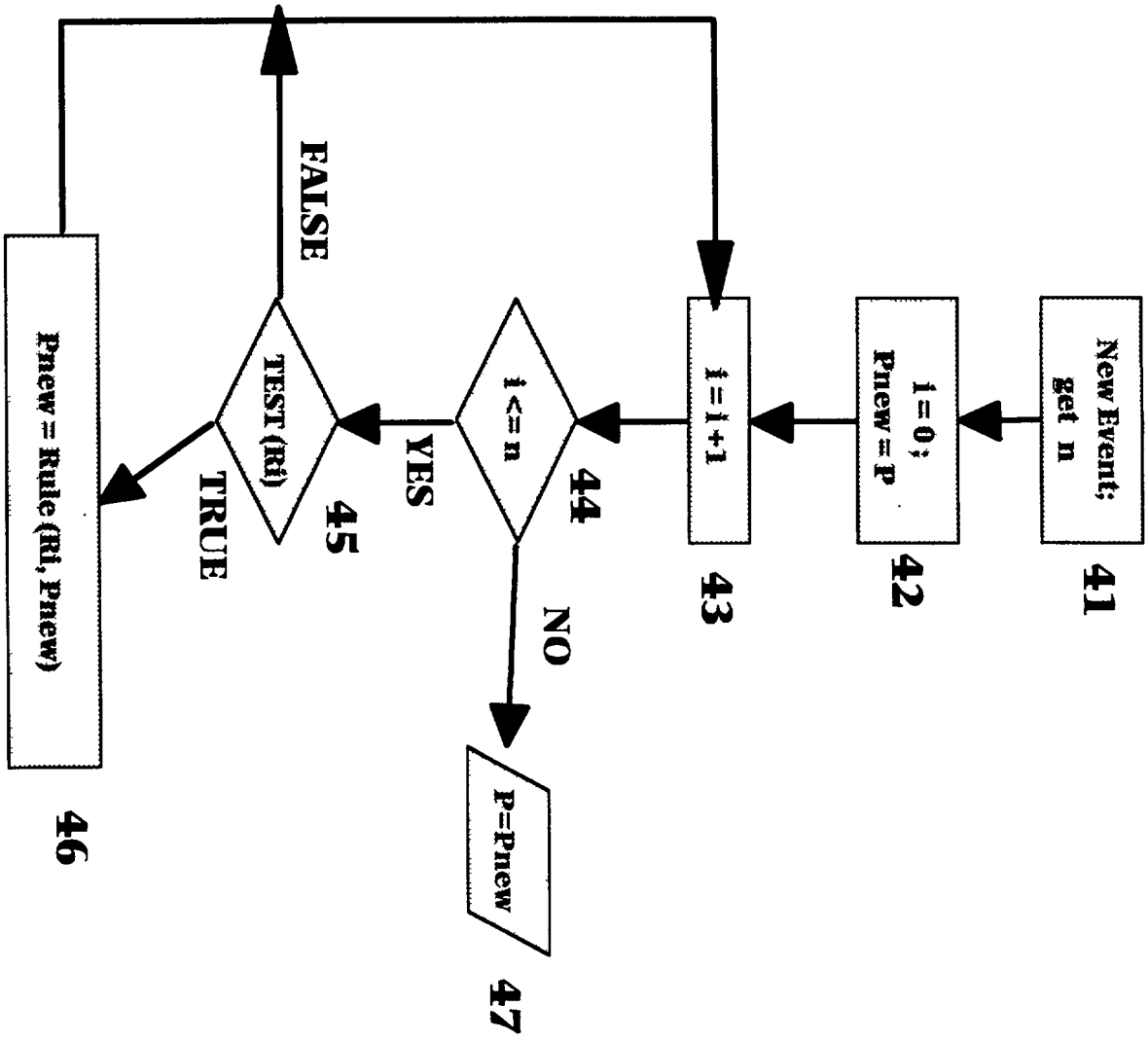

New Event;
get n
41
i = 0 ;
Pnew = P
42
i = i +1
43
i <= n
44
YES
NO
TEST (Ri)
45
TRUE
FALSE
Pnew = Rule (Ri, Pnew)
46
P=Pnew
47

FIG. 4

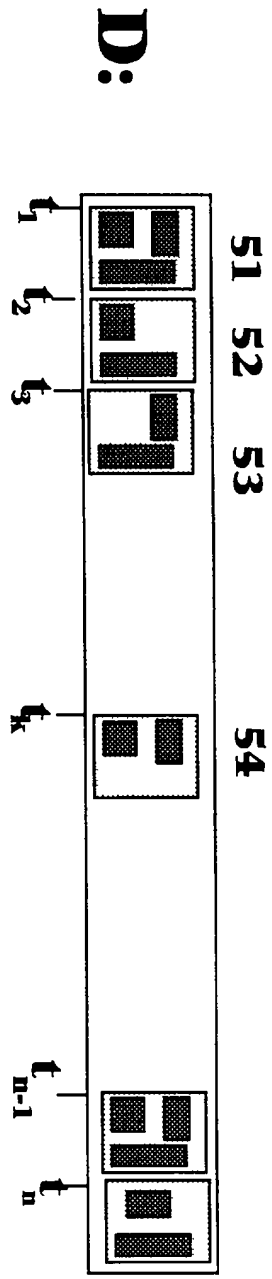


FIG. 5

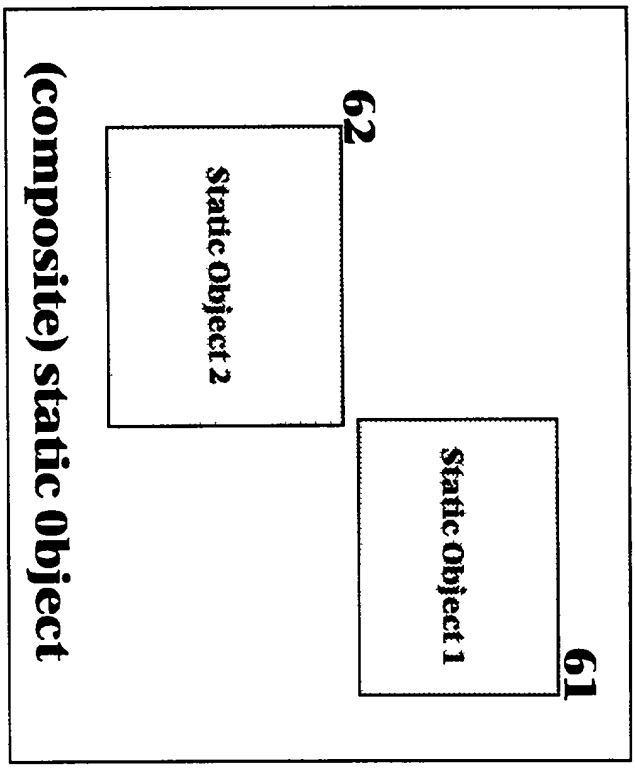

(composite) static Object
Static Object 2
62
Static Object 1
61

FIG. 6

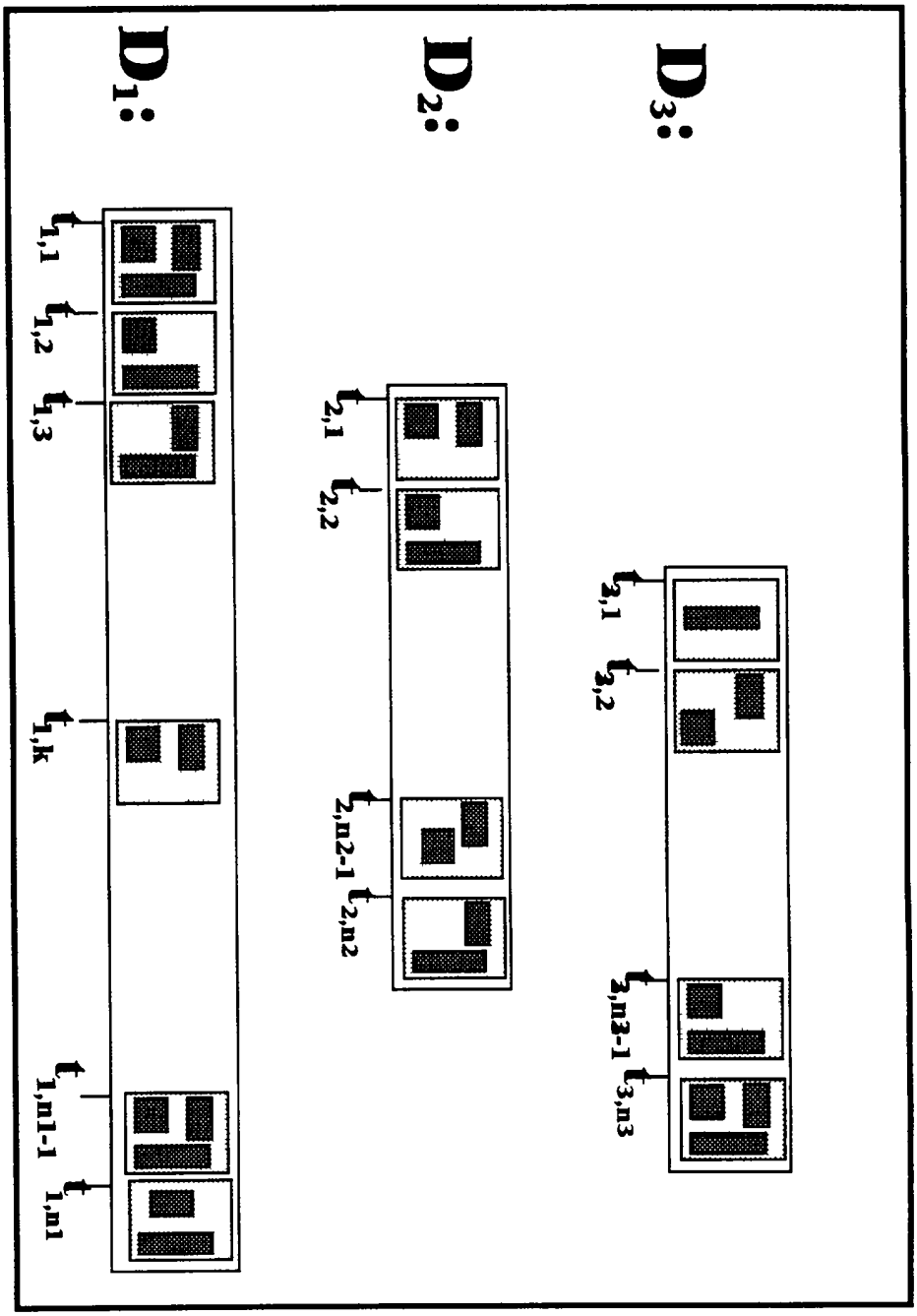
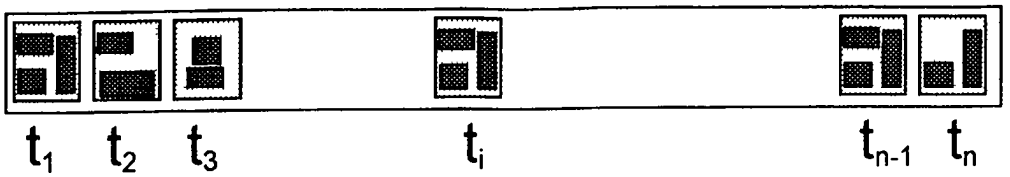
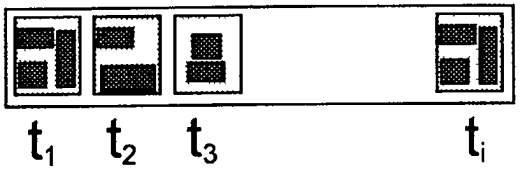

D3:
t2,1
t2,2
t3,n3-1
t3,n3
D2:
t2,1
t2,2
t2,n2-1
t2,n2
D1:
t1,1
t1,2
t1,3
t1,k
t1,n1-1
t1,n1

FIG. 7

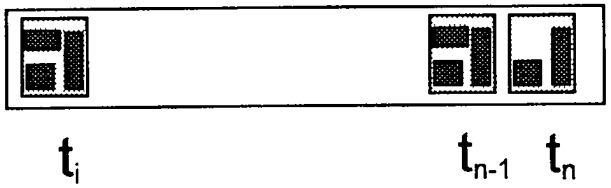
D :



81
Head (D,i):



82
Tail (D, i):



83
Delay (D, t) :

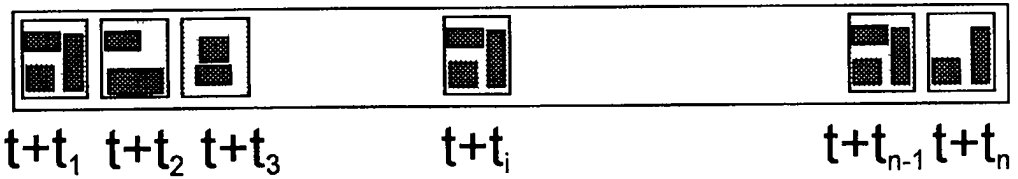


FIG. 8

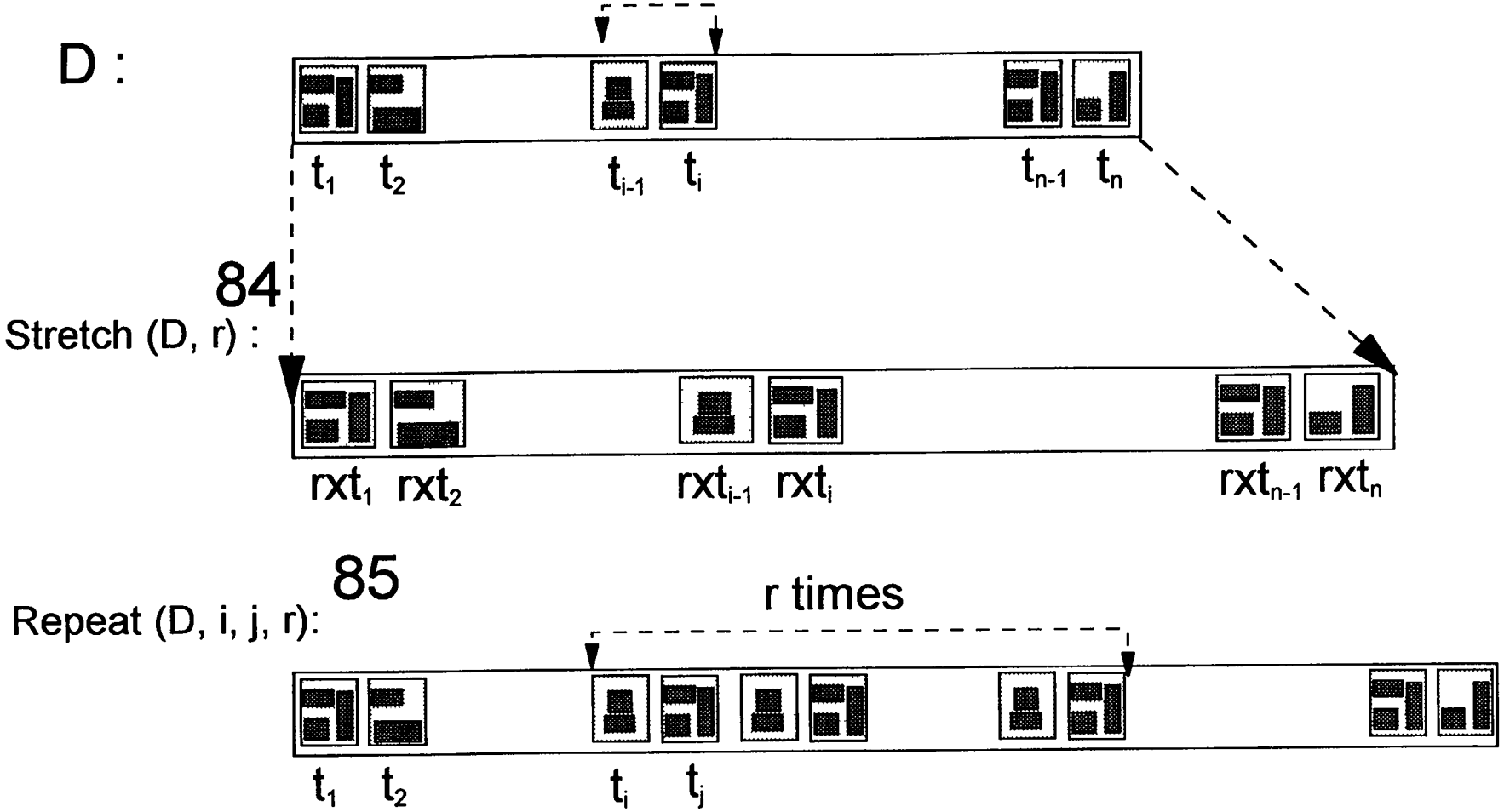

D :
t_1 t_2 t_{i-1} t_i t_{n-1} t_n
84
Stretch (D, r) :
rxt_1 rxt_2 rxt_{i-1} rxt_i rxt_{n-1} rxt_n
85
Repeat (D, i, j, r):
r times
t_1 t_2 t_i t_j

FIG. 8A

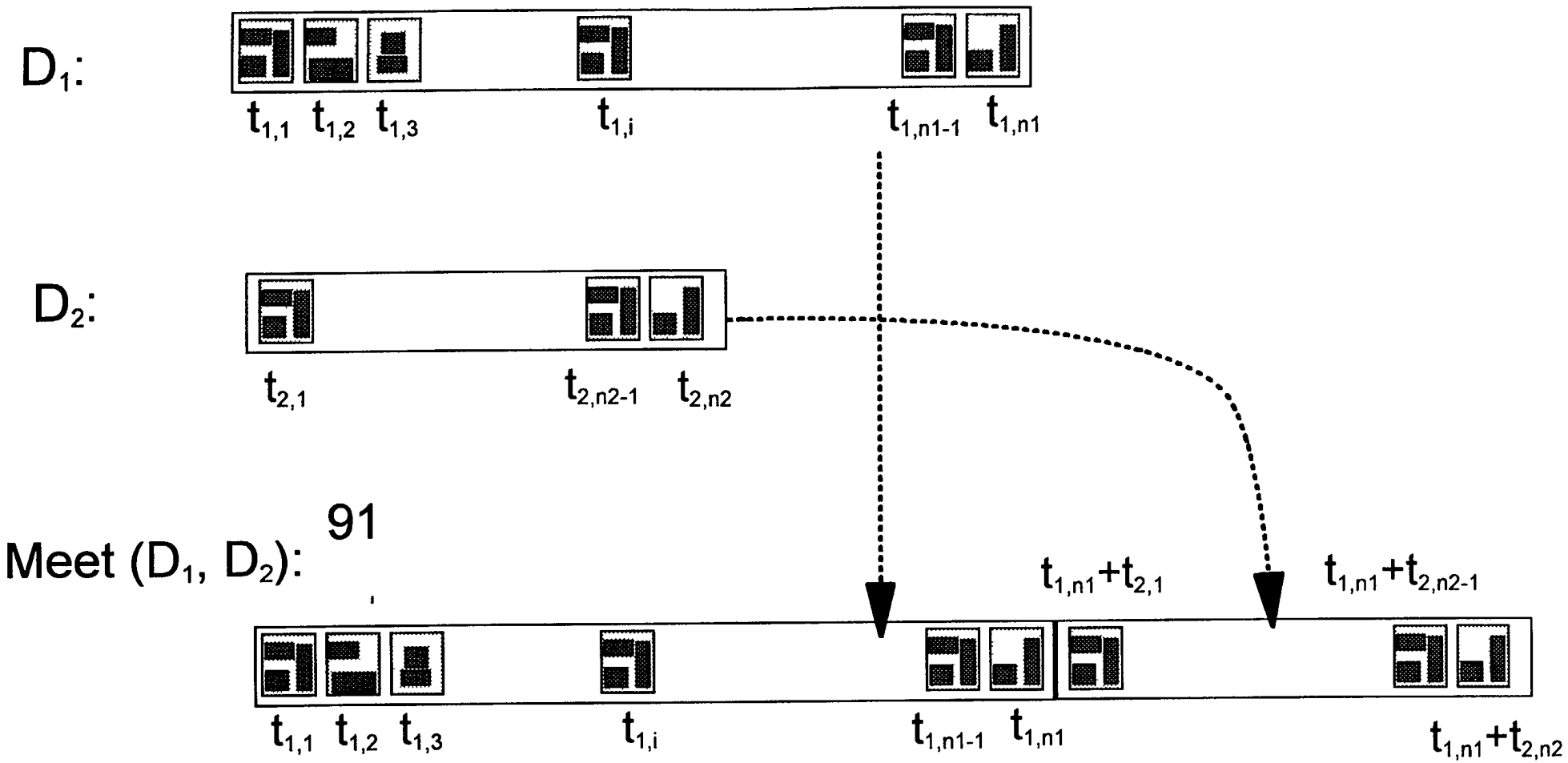
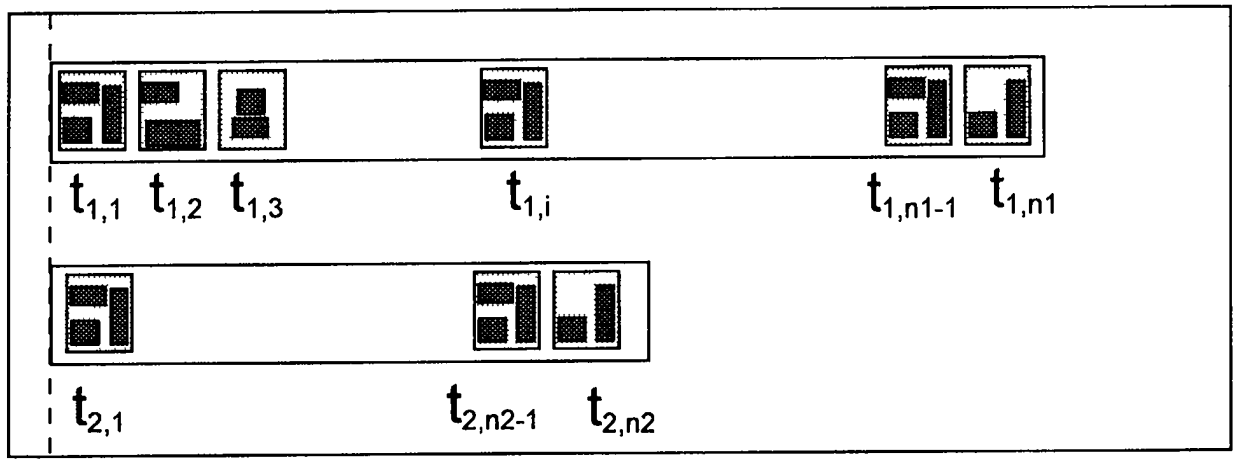

D1:
t1,1 t1,2 t1,3 t1,i t1,n1-1 t1,n1
D2:
t2,1 t2,n2-1 t2,n2
Meet (D1, D2):
91
t1,n1+t2,1
t1,n1+t2,n2-1
t1,1 t1,2 t1,3 t1,i t1,n1-1 t1,n1
t1,n1+t2,n2

FIG. 9

Co-start(D_1,D_2): 92



Co-end(D_1, D_2): 93

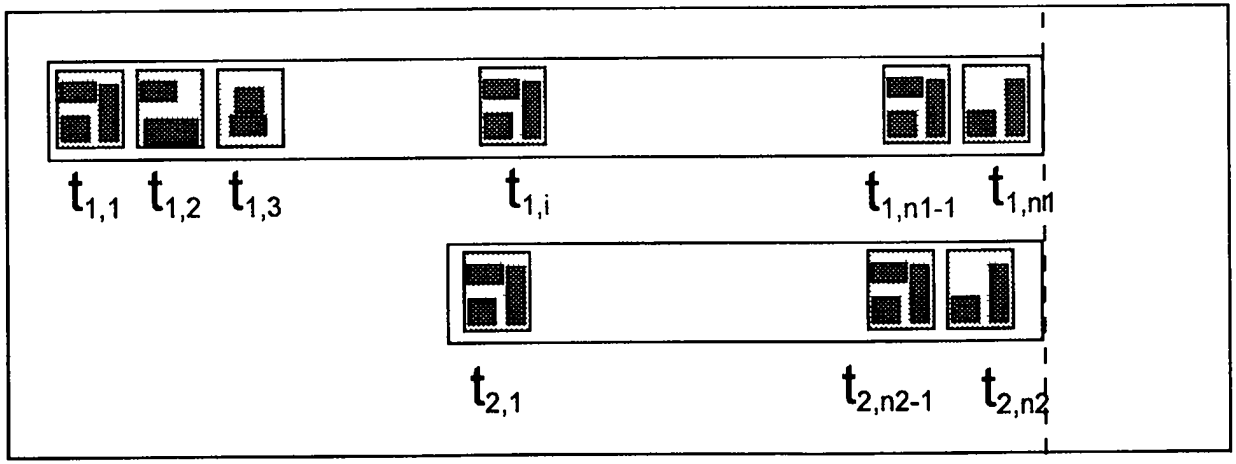
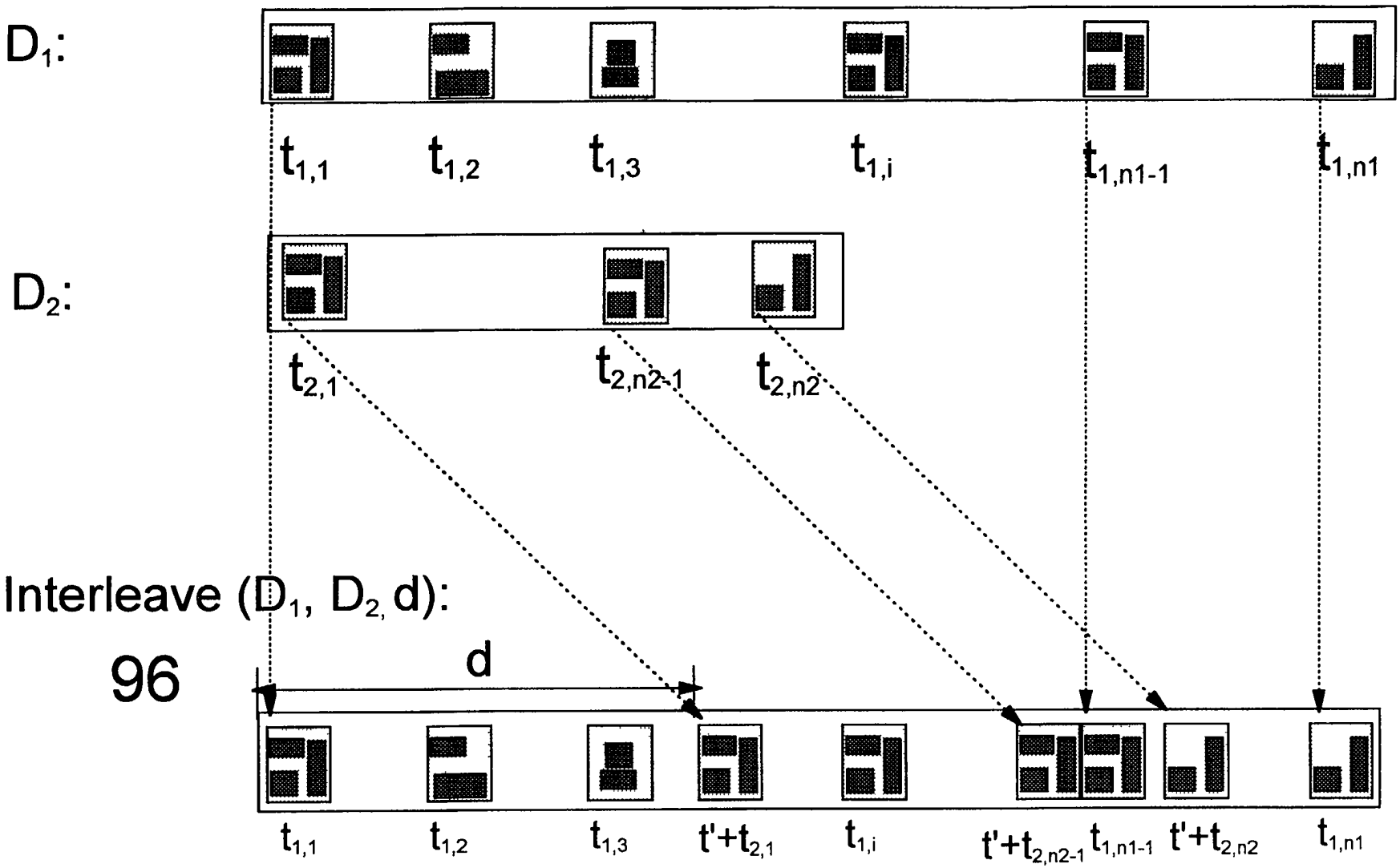


FIG. 9A


D1:
t1,1
t1,2
t1,3
t1,i
t1,n1-1
t1,n1
D2:
t2,1
t2,n2-1
t2,n2
Interleave (D1, D2, d):
96
d
t1,1
t1,2
t1,3
t'+t2,1
t1,i
t'+t2,n2-1
t1,n1-1
t'+t2,n2
t1,n1

$$t' = d + t_{1,1}$$

FIG. 9B

"EXPRESS MAIL" LABEL NO. EM526133507
DATE OF DEPOSIT: April 30, 1998

Docket No. YO998-137

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name;

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled:

SYSTEM AND METHOD FOR PROGRAMMATIC GENERATION OF CONTINUOUS MEDIA PRESENTATIONS
the specification of which (check one)

_X_ is attached hereto.

___ was filed on _____ as United States Application Number

or PCT International Application Number _____

and was amended on _____ (if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119(a)-(d) or §365(b) of any foreign application(s) for patent or inventor's certificate, or §365(a) of any PCT International application which designated at least one country other than the United States, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate, or PCT International application, having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)			Priority Claimed
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	___ Yes ___ No
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	___ Yes ___ No
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	___ Yes ___ No

I hereby claim the benefit under 35 U.S.C. §119(e) of any United States provisional application(s) listed below.

_____ (Application Number)	_____ (Filing Date)
_____ (Application Number)	_____ (Filing Date)

I hereby claim the benefit under 35 U.S.C. §120 of any United States Application(s), or §365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States, or PCT International application in the manner provided by the first paragraph of 35 U.S.C. §112, I acknowledge the duty to disclose information material to the patentability of this application as defined in 37 CFR §1.56 which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

_____ (Application Serial No.)	_____ (Filing Date)	_____ (Status) (patented, pending, abandoned)
_____ (Application Serial No.)	_____ (Filing Date)	_____ (Status) (patented, pending, abandoned)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: As a named inventor I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith (list name and registration number).

Manny W. Schecter (Reg. 31,722), Terry J. Ilardi (Reg. 29,936), Christopher A. Hughes (Reg. 26,914), Edward A. Pennington (Reg. 32,588), John E. Hoel (Reg. 26,279), Joseph C. Redmond, Jr. (Reg. 18,753), and David M. Shofi (Reg. 39,835)

Send Correspondence to: David M. Shofi, Intellectual Property Law

IBM Corporation, P.O. Box 218, Yorktown Heights, New York 10598

Direct Telephone Calls to: (name and telephone number) David M. Shofi (914) 945-3247

Asit Dan
Full name of sole or first inventor

Inventor's Signature _____ Date 4/30/98

75 Ginsborg Avenue, West Harrison, New York 10604
Residence

India
Citizenship

same as above
Post Office Address

Docket No. YO998-137

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

Jai Prakash Menon
Full name of second joint-inventor, if any

Inventor's signature — Date: 30 APRIL 1998

55 Campus Road, Peekskill, New York 10566
Residence

India
Citizenship

same as above
Post Office Address

Junehwa Song
Full name of third joint inventor, if any

Inventor's Signature — Date: 30 April 1998

200 Diplomat Drive, Apt. 5F, Mt. Kisco, New York 10549
Residence

Korea
Citizenship

same as above
Post Office Address

Full name of fourth joint-inventor, if any

Inventor's signature — Date

Residence

Citizenship

Post Office Address

Full name of fifth joint inventor, if any

Inventor's Signature — Date

Residence

Citizenship

Post Office Address

Full name of sixth joint-inventor, if any

Inventor's signature — Date

Residence

Citizenship

Post Office Address

Full name of seventh joint-inventor, if any

Inventor's signature — Date

Residence

Citizenship